TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/881,505
Filing Date	6/14/2001
First Named Inventor	Bipul Binit Sinha, et al.
Art Unit	2161
Examiner Name	LeRoux, Etienne Pierre
Attorney Docket Number	oracle01.016
Total Number of Pages in This Submission	12

ENCLOSURES *(Check all that apply)*

- [ ] Fee Transmittal Form
  - [ ] Fee Attached
- [ ] Amendment/Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [ ] Reply to Missing Parts/ Incomplete Application
  - [ ] Reply to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) ____________
  - [ ] Landscape Table on CD
- [ ] After Allowance Communication to TC
- [ ] Appeal Communication to Board of Appeals and Interferences
- [x] Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- [ ] Proprietary Information
- [ ] Status Letter
- [ ] Other Enclosure(s) (please Identify below):

Remarks

Response to Notice of Non-Compliant Appeal Brief
Attn: Bridget C. Monroe

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	Gordon E. Nelson, Patent Attorney PC		
Signature	/Gordon E. Nelson/		
Printed name	Gordon E. Nelson		
Date	July 14, 2006	Reg. No.	30,093

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

Signature	/Gordon E. Nelson/		
Typed or printed name	Gordon E. Nelson	Date	July 14, 2006

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
(oracle01.016)

Applicant: Bipul Binit Sinha et al. **Paper No.:**

Application No: 09/881,505 **Group Art Unit:** 2161

Filed: 6/14/01 **Examiner:** Bridget C. Monroe
Patent Appeals Specialist

Title: *Two-stage commit with queryable caches*

Commissioner for Patents
Alexandria, VA 22313-1450

Response to Notification of Non-Compliant Appeal Brief, 37 CFR 41.37

Appellant received a Notification of Non-Compliant Appeal Brief in the above application dated July 5, 2006. As requested in item 10 of the Notification, appellant is submitting a replacement section (5) Summary of the Claimed Subject Matter instead of an entire new brief per MPEP 1205.03. Included are a clean copy of the replacement section (5) and a copy in which changes from the previous section (5) are marked.

No fees are believed to be required for this response. If any are, please charge them to deposit account number 501315.

Respectfully submitted,

/Gordon E. Nelson/
Attorney of record,
Gordon E. Nelson
57 Central St., P.O. Box 782
Rowley, MA, 01969,
Registration number 30,093
Voice: (978) 948-7632
Fax: (866) 723-0359
July 14, 2006
Date

Clean copy of Replacement Section (5)

(5) Summary of claimed subject matter

The independent claims in this application are claims 5, 9 10, 11, 22, 26, 30, 31. All are addressed to the same subject matter, namely a technique for optimizing protocols used in distributed systems to ensure that all of the components of a distributed system that are affected by a transaction remain consistent with regard to the transaction. A preferred embodiment of the invention is described beginning at page 15, line 14. FIG. 3 and page 15, line 14-17, line 10 describe describes one of these protocols, the well-known two-phase commit protocol; page 17, lines 11-24 describe a prior-art optimization of the two-phase commit protocol. As may be seen there, the protocols achieve consistency among the components that are affected by the transaction by ensuring that a transaction that changes the state of the affected components either changes the state of all of the affected components or changes the state of none of them. The later situation occurs if one or more of the components is unable to commit the transaction.

Each of the components affected by the transaction is either a *coordinator* or a *cohort* with regard to the protocol. The coordinator sends a *request commit* message to the cohorts which indicates to the cohorts that a state change resulting from a transaction is about to take place; each cohort replies to the *request commit* message with an *agree* message if the cohort can make the state change or an *abort* message if it cannot. If the coordinator receives *agree* messages from all of the cohorts, the coordinator sends a *commit* message to each of the cohorts and the cohorts respond to the *commit* message by making the state change and sending an *acknowledgement* message to the coordinator. When the coordinator has received *acknowledgment* messages from all of the cohorts, it makes the change itself. If the coordinator receives an *abort* message from any of the cohorts, it does not make the change itself and sends an *abort* message to each of the cohorts. The cohorts respond to the *abort* message by not making the change.

The optimization that is the subject matter of independent claims 5, 9, 10, 11, 22, 26, 30, and 31 takes advantage of the fact that a component of the distributed system may be read only with regard to the transaction. If it is, the transaction will not affect the component's state and the component need not participate in the protocol. Indeed, if the coordinator

knows that a cohort is read only with regard to the transaction, it need not determine whether the cohort can make the state change. As is clear from the foregoing, what is needed is a way of making it possible for the coordinator to know *at the time it sends the request commit messages to the cohorts* whether a cohort is read only with regard to the transaction.

Claim 11

The technique for making it possible for the coordinator to know at the time it sends the request commit messages to the cohorts whether a cohort is read only with regard to the transaction is set forth broadly in method form in claim 11, which claims the technique from the point of view of the coordinator. Claim 11 reads as follows. Reference numbers refer to FIG. 4:

> **11.** A method practiced in a first component of a distributed system that exchanges messages (403) belonging to a transaction with one or more other components of the distributed system of optimizing a protocol, the protocol being employed by the first component and the other component in making the transaction, the first component being a coordinator for the protocol, and
>
> the method comprising the steps of:
>
> receiving an augmented one of the messages (401) from the other component, the other component having augmented the message by adding protocol state information (405) to the message, the protocol state information indicating a state of the other component that is relevant to the protocol;
>
> retaining the state of the other component indicated in the augmented message (413); and
>
> using the retained state to optimize the protocol.

In the two-phase commit protocol, which is a species of what is claimed above, the protocol state information is that the cohort is read only and the mechanism set forth in the claim for ensuring that the coordinator always knows whether a transaction is read-only with regard to a cohort is described at page 17, line 26-page 18, line 30. The cohorts *augment* the messages 403 belonging to the transaction with protocol state information 405 indicating a state of the cohort that is relevant to the protocol, in this case that the cohort is read only with respect to the transaction. The coordinator keeps track of the

state for each cohort as shown at 413, updating a cohort's state as augmented messages 403 come in from the cohort. When it is time to change state, the coordinator can use the retained state for the cohorts to optimize the protocol that is performed when the state changes.

FIG. 5 is a flowchart for using augmented messages 403 indicating (407) whether a cohort is read only with regard to a transaction to optimize the two-phase commit protocol. The flowchart is described at page 19, lines 12-31. The claimed step of "receiving an augmented one of the messages is shown at 515; the step of "retaining the state" is shown at 517; the step of "using the retained state to optimize the protocol" (here, the two-phase commit protocol) is shown at 519, 521, 527, and 529. If retained state 413 indicates that a cohort is read-only with regard to the transaction, the coordinator simply sends the cohort a 2-phase commit "abort" message and doesn't bother with the remainder of the 2-phase commit for that cohort.

The other independent claims

FIGs. 4, 5, and 6 and the portions of the Specification beginning at page 17, line 26 in which these figures are discussed disclose the inventions of all of the independent claims. A discussion of generalizations of the optimization technique may be found beginning at page 21, line 6.

Claim 5

Claim 5 is a generic method claim which claims the optimization technique from the point of view of the cohort. The flowchart for the cohort is shown at FIG. 6 and described at page 20, lines 1-12. When a transaction message is to be sent to the coordinator (605), the cohort determines whether it is read only (607); if it is, it sets bit 407 to read only (621); otherwise, it sets bit 407 to not read only (623) and sends augmented message 401 with the set bit 407 to the coordinator (625).

Claims 9 and 10

Method claims 9 and 10 are addressed to the species of the generic methods that are

represented by the optimization of the 2-phase commit protocol; claim 9 is directed to the optimized 2-phase commit protocol as claimed from the point of view of the coordinator (Fig. 5); claim 10 is directed to the optimized 2-phase commit protocol as claimed from the point of view of the cohort (Fig. 6).

Claims 22, 26, 30 and 31

Independent claims 22, 26, 30, and 31 are apparatus claims that are addressed to coordinators and cohorts which employ the technique to optimize protocols.

- *claim 22* is to a generic coordinator which employs the technique: (Fig. 5);
- *claim 26* is to a generic cohort which does so (Fig. 6);
- *claim 30* is to a species of the coordinator which employs the technique to optimize the two-phase commit protocol (Fig. 5); and
- *claim 31* is to a species of the cohort which employs the technique to optimize that protocol (Fig. 6).

Copy of Replacement of Section (5) showing changes

(5) Summary of claimed subject matter

The independent claims in this application are claims 5, ~~8,~~9 10, 11, 22, ~~24~~26, 30, 31. All are addressed to the same subject matter, namely a technique for optimizing protocols used in distributed systems to ensure that all of the components of a distributed system that are affected by a transaction remain consistent with regard to the transaction. A preferred embodiment of the invention is described beginning at page 15, line 14. FIG. 3 and page 15, line 14-17, line 10 describe describes one of these protocols, the well-known two-phase commit protocol; page 17, lines 11-24 describe a prior-art optimization of the two-phase commit protocol. As may be seen there, the protocols achieve consistency among the components that are affected by the transaction by ensuring that a transaction that changes the state of the affected components either changes the state of all of the affected components or changes the state of none of them. The later situation occurs if one or more of the components is unable to commit the transaction.

Each of the components affected by the transaction is either a *coordinator* or a *cohort* with regard to the protocol. The coordinator sends a *request commit* message to the cohorts which indicates to the cohorts that a state change resulting from a transaction is about to take place; each cohort replies to the *request commit* message with an *agree* message if the cohort can make the state change or an *abort* message if it cannot. If the coordinator receives *agree* messages from all of the cohorts, the coordinator sends a *commit* message to each of the cohorts and the cohorts respond to the *commit* message by making the state change and sending an *acknowledgement* message to the coordinator. When the coordinator has received *acknowledgment* messages from all of the cohorts, it makes the change itself. If the coordinator receives an *abort* message from any of the cohorts, it does not make the change itself and sends an *abort* message to each of the cohorts. The cohorts respond to the *abort* message by not making the change.

The optimization that is the subject matter of independent claims 5, ~~8~~9, 10, 11, 22, ~~24~~26, 30, and 31 takes advantage of the fact that a component of the distributed system may be read only with regard to the transaction. If it is, the transaction will not affect the component's state and the component need not participate in the protocol. Indeed, if the

coordinator knows that a cohort is read only with regard to the transaction, it need not determine whether the cohort can make the state change. As is clear from the foregoing, what is needed is a way of making it possible for the coordinator to know *at the time it sends the request commit messages to the cohorts* whether a cohort is read only with regard to the transaction.

Claim 11

The technique for making it possible for the coordinator to know at the time it sends the request commit messages to the cohorts whether a cohort is read only with regard to the transaction ~~doing this~~ is set forth broadly in method form in claim 11, which claims the technique from the point of view of the coordinator. Claim 11 ~~which~~ reads as follows. Reference numbers refer to FIG. 4:

> **11.** A method practiced in a first component of a distributed system that exchanges messages (403) belonging to a transaction with one or more other components of the distributed system of optimizing a protocol, the protocol being employed by the first component and the other component in making the transaction, the first component being a coordinator for the protocol, and
>
> the method comprising the steps of:
>
> receiving an augmented one of the messages (401) from the other component, the other component having augmented the message by adding protocol state information (405) to the message, the protocol state information indicating a state of the other component that is relevant to the protocol;
>
> retaining the state of the other component indicated in the augmented message (413); and
>
> using the retained state to optimize the protocol.

In the two-phase commit protocol, which is a species of what is claimed above, the protocol state information is that the cohort is read only and the mechanism set forth in the claim for ensuring that the coordinator always knows whether a transaction is read-only with regard to a cohort is described at page 17, line 26-page 18, line 30. The cohorts *augment* the messages 403 belonging to the transaction with protocol state information 405 indicating a state of the cohort that is relevant to the protocol, in this case that the cohort is read only with respect to the transaction. The coordinator keeps track of the

state for each cohort as shown at 413, updating a cohort's state as augmented messages 403 come in from the cohort. When it is time to change state, the coordinator can use the retained state for the cohorts to optimize the protocol that is performed when the state changes.

FIG. 5 is a flowchart for using augmented messages 403 indicating (407) whether a cohort is read only with regard to a transaction to optimize the two-phase commit protocol. The flowchart is described at page 19, lines 12-31. The claimed step of "receiving an augmented one of the messages is shown at 515; the step of "retaining the state" is shown at 517; the step of "using the retained state to optimize the protocol" (here, the two-phase commit protocol) is shown at 519, 521, 527, and 529. If retained state 413 indicates that a cohort is read-only with regard to the transaction, the coordinator simply sends the cohort a 2-phase commit "abort" message and doesn't bother with the remainder of the 2-phase commit for that cohort.

The other independent claims

<u>FIGs. 4, 5, and 6 and the portions of the Specification beginning at page 17, line 26 in which these figures are discussed disclose the inventions of all of the independent claims. A discussion of generalizations of the optimization technique may be found beginning at page 21, line 6.</u>

<u>Claim 5</u>

Claim 5 is a generic method claim which claims the optimization technique from the point of view of the cohort. The flowchart for the cohort is shown at FIG. 6 and described at page 20, lines 1-12. When a transaction message is to be sent to the coordinator (605), the cohort determines whether it is read only (607); if it is, it sets bit 407 to read only (621); otherwise, it sets bit 407 to not read only (623) and sends augmented message 401 with the set bit 407 to the coordinator (625).

<u>Claims 9 and 10</u>

Method claims 9 and 10 are addressed to the species of the generic methods that are

represented by the optimization of the 2-phase commit protocol; claim 9 is directed to the optimized 2-phase commit protocol as claimed from the point of view of the coordinator~~;~~ (Fig. 5); claim 10 is directed to the optimized 2-phase commit protocol as claimed from the point of view of the cohort (Fig. 6).

Claims 22, 26, 30 and 31

Independent claims 22, 26, 30, and 31 are apparatus claims that are addressed to coordinators and cohorts which employ the technique to optimize protocols.

- *claim 22* is to a generic coordinator which employs the technique: (Fig. 5);
- *claim 26* is to a generic cohort which does so (Fig. 6);
- *claim 30* is to a species of the coordinator which employs the technique to optimize the two-phase commit protocol (Fig. 5); and
- *claim 31* is to a species of the cohort which employs the technique to optimize that protocol (Fig. 6).

~~FIGs. 4, 5, and 6 and the portions of the Specification beginning at page 17, line 26 in which these figures are discussed disclose the inventions of all of the independent claims. A discussion of generalizations of the optimization technique may be found beginning at page 21, line 6.~~